

The ToF CMOS Visual Cortex Project

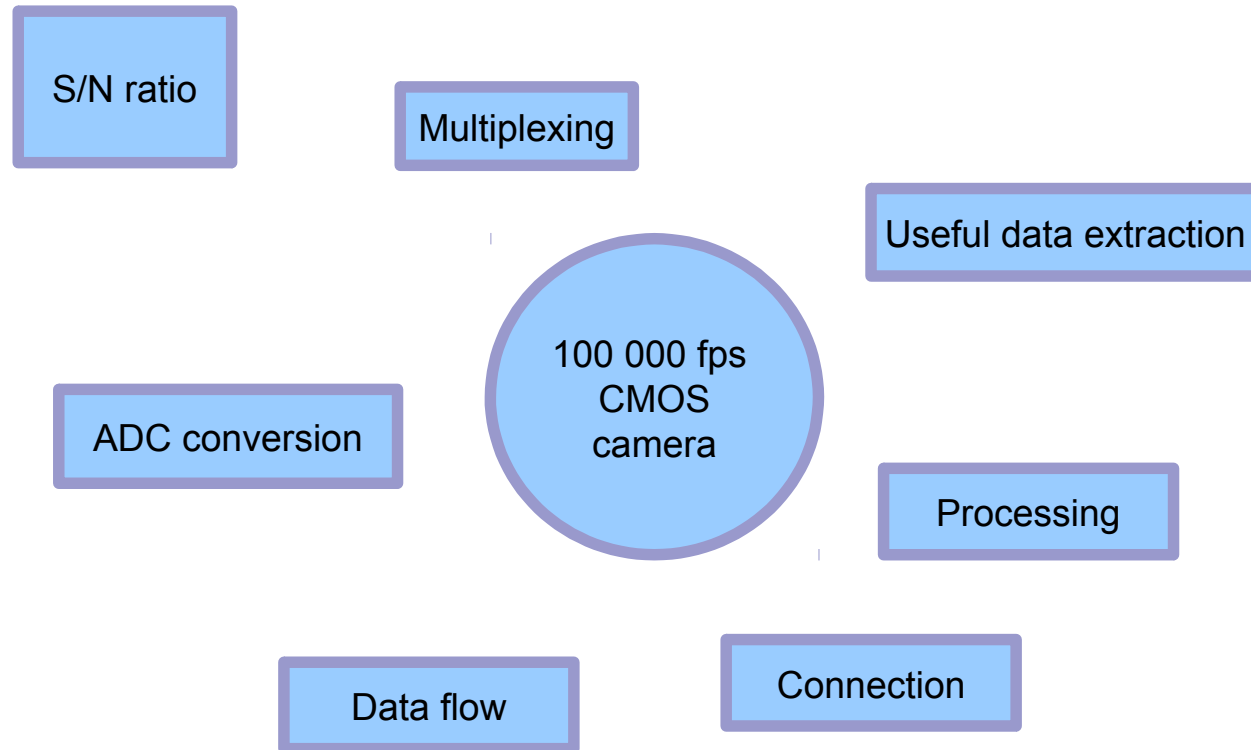
P. Van Esch¹, P. Mutti¹, M. Platz¹, E. Ruiz-Martinez¹

¹ Institut Laue-Langevin, Grenoble, France

Concept

- Thermal neutron detection
 - Imaging versus counting
 - ToF: only counting: no camera is fast enough
 - Sustained time resolution ~ a few microseconds
- CMOS camera
 - Sensor has potential to be fast enough
 - Not exploited yet
- Propose: CMOS camera with 100 000 fps sustained rate
- Treat information online and only extract useful information
(visual cortex principle)

Challenges



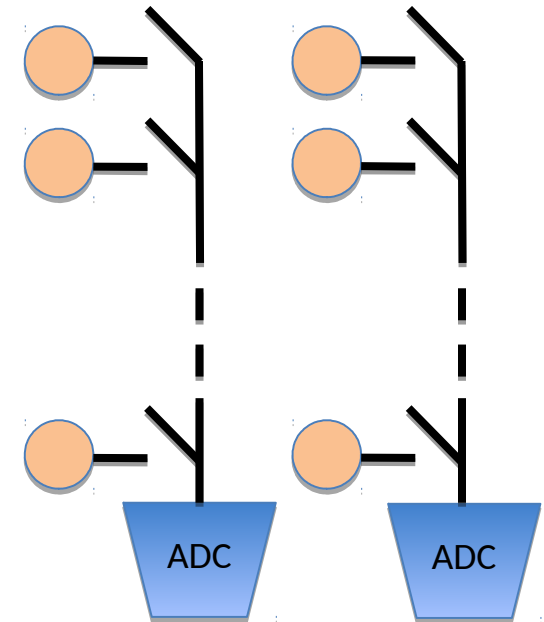
Front-end challenges

Actual CMOS:

- 2000 pixels multiplexed per column
- Wilkinson-type “cheap” ADC converter per column
- ~ 200 000 Hz sampling

Proposition:

- Flash ADC, ~200 MHz sampling
- Faster multiplexing vs S/N ratio (pixel filter ?)
- ADC probably *too hot*, must be off-sensor
- Challenge: get 2000 analogue signals off-chip



Data Flow challenge

Data flow challenge

- 2000 ADC at 200 MHz at 16 bit = **800 GB/s of raw image data!**
- Impossible (and unreasonable) to store
- Difficult to transport

Compare ATLAS: 200 MB/s
After trigger



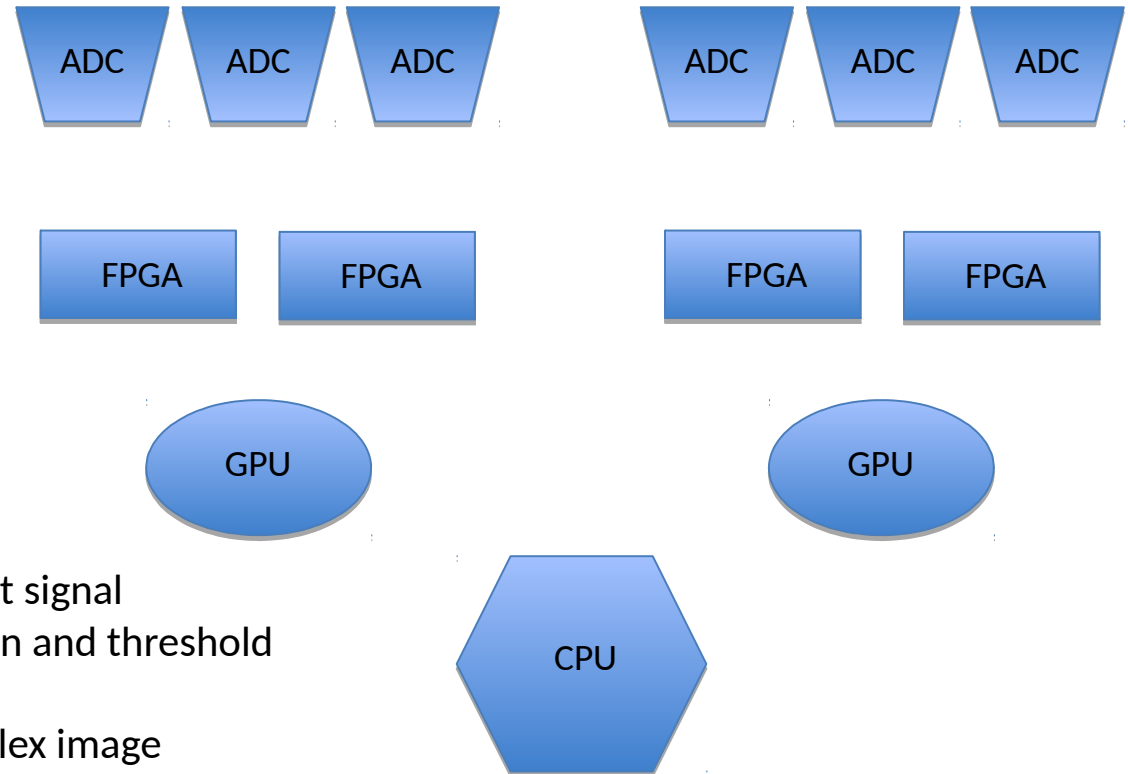
Processing challenge

Processing challenge

- 400 000 000 000 pixels to be treated per second
- Centralized processing impossible
- Advanced distributed image feature recognition needed



Layered processing



Layered processing

- Nearby ADC to FPGA for first level fast signal treatment, mainly hot pixels reduction and threshold discrimination
- FPGA to several GPUs for more complex image manipulation/correction e.g. pattern recognition, calibration, further specific noise reduction
- CPU at the end for final data composition and handling of the data stream to external acquisition

Impact

- Imagine we have a camera at 100 000 fps
- Thermal neutron ToF (was the onset)
 - Pulsed sources
- Medical/biology: fast kinetic phenomena, such as drug intake at the cellular level
- Industry: onset of material ruptures
- Everywhere one cannot use stroboscopy, nor a trigger known in advance (but determined by the image itself)